Real-Time Patient Action Detection Using Machine Learning

Dr.Shabeen Taj G A[1], Shivukumar2, Chandana H M[3], Anusha N[4,] Surendra S[5]
[1]*Assistant professor, Department of CS&E, Government Engineering College, Ramanagara*
[1,3,4,5]*Student, Department of CS&E, Government Engineering College, Ramanagara*

***Abstract*—Action detection using machine learning (ML) is an emerging research area that involves automatically recognizing and localizing human actions in videos. In this approach, a model is trained on a large dataset of videos, which learns to identify patterns and features that are unique to each action. Once trained, the model can then be used to detect actions in new, unseen videos. The key challenge in action detection is to accurately identify the start and end times of each action, as well as distinguish between different actions that may occur simultaneously. In recent years, significant progress has been made in developing ML-based action detection algorithms, which have found applications in diverse fields, such as patient surveillance, sports analysis, and human-computer interaction. This project provides an overview of the state-of-the-art techniques in action detection using ML, including the datasets, models, and evaluation metrics commonly used in this field. Patient action detection using Machine Learning (ML) is a technique used to automatically recognize and categorize patient activities in healthcare settings. This technique utilizes sensor data from video cameras to detect activities such as Normal, headache, leg pain, and heart pain. ML algorithms are used to analyze this sensor data and recognize specific patterns associated with each activity. This approach can provide important information to healthcare providers, allowing them to monitor patient activities, assess their health status, and identify potential problems. In this abstract, we discuss the importance of patient action detection in healthcare, the challenges involved in implementing ML-based techniques, and the potential benefits of this approach.**

***Index Terms*—Human Action recognition, Classification, Radom Forest Model, Supervised Machine learning.**

I. INTRODUCTION

In today's world, smart devices such as smart phones, smart watches, and smart clothing have become widely used. These devices are equipped with various sensors such as cameras, gyroscopes, and accelerometers, which are constantly being researched and developed for the field of human activity recognition. Accelerometers and gyroscopes are commonly used to measure the acceleration and angular velocity of subjects in this area of research. There are several existing techniques for recognizing activities, and sensor-based methods are generally considered superior to computer vision-based methods. Vision-based methods, such as using video cameras in monitoring zones, can be inconvenient as they require maintenance attention, are sensitive to environmental noise, costly, and unproductive in terms of power consumption. In contrast, sensor-based human activity tracking is considered a more efficient approach as it does not have the limitations of vision-based techniques. In this approach, motions of the human body are tracked using sensors attached to the body or built-in Smartphone sensors, and data obtained from the inertial measurement unit is processed. Due to these advantages, researchers have explored the possibility of using sensor networks and classification algorithms for human activity recognition (HAR) to classify different actions of users. The main objective of this work is to identify human activities in various environments, including detecting immoral activities. Our proposed approach uses CNN, SVM, and random forest as classifiers for recognizing activities in HAR problems. The Convolutional Neural Network (CNN) has proven to be highly effective as a fundamental deep learning model for activity recognition in various fields. Our model focuses on activity recognition in different environments and has been tested in an experiment involving 36 users performing six physical activity patterns, such as walking, jogging, standing, sitting, going upstairs, going downstairs, playing guitar, and sports. For activity recognition, we primarily utilize the accelerometer sensor in smart phones to measure acceleration. The data collected from the accelerometer sensor is used for this purpose. Since there is no direct method to measure acceleration in each spatial dimension, the tri-axial accelerometer embedded in modern smart phones is used. The validation task involves acquiring data from these smart phone sensors. Human activity recognition (HAR) involves the identification of various physical activities performed by individuals in different environments, such as sports, jogging, playing guitar, etc. This process typically involves training a supervised learning model using input data from cameras, and then displaying the recognized activity based on the input received. HAR is a procedure aimed at accurately identifying and classifying different human activities. Our goal is to recognize the human activity in real time with patient admitted in the hospital and device should said the activity performed by human, we use the camera to capture the video. Basically there is none or less system available in the market to detect the patient moment detection in hospital. It detects the certain activities done by the patients and try to alert the medical team for the immediate response. There are many cases happening around in the hospital where patients are not monitored 24 * 7. Only in ICU they are monitoring every time, but sometimes in general ward also patients are falling sick and medical teams cannot track each and every time medical team cannot track patients every time. So we developed a system which detects the patient activities and notifies if required. Our system detect the single patient activities on camera like chest pain, headache or leg pain , stomach pain which is very much important to be notified by the medical team to take action as soon as possible.

II. RELATED WORK

According to basic survey we have found few related works on human activity recognition in healthcare domain some are [1] “Deep Learning for Human Activity Recognition in Healthcare Contexts” by Alaa Halawani published in the year 2019 which basically elaborates on deep learning-based approach for recognizing human activities in healthcare settings. The authors trained a convolutional neural network (CNN) on accelerometer data to detect activities such as walking, standing, and lying down. The study showed promising results for activity recognition in healthcare contexts.[2] “Smart Homes for Elderly Healthcare—Recent Advances and Research Challenges” by Chen Song published in the yea 2020 states an overview of recent advances in smart homes for elderly healthcare. The authors discuss the use of sensors and machine learning algorithms to detect falls, monitor vital signs, and track daily activities. The paper also highlights the challenges of developing smart homes for healthcare and future research directions.[3] “Patient Action Prediction with Recurrent Neural Networks for Smart Hospitals” by Zhang published in the year 2018 explains a recurrent neural network (RNN) approach for predicting patient actions in a smart hospital setting. The authors used RNNs to model the temporal dependencies in patient actions and achieved high accuracy in predicting actions such as medication intake, bathroom visits, and meal times.[4] Activity Recognition and Abnormality Detection with the Multimodal Sensor Data in a Nursing Home by Sung Bum Kim published in the year 2019 explains a multimodal sensor-based approach for activity recognition and abnormality detection in a nursing home. The authors used machine learning algorithms to classify activities such as sitting, standing, and walking, and detected abnormal events such as falls and wandering.[5] A Survey of Deep Learning Techniques for Human Activity Recognition by Khan published in the year 2020 states on a comprehensive survey of deep learning techniques for human activity recognition. The authors review the state-of-the-art deep learning models for activity recognition, including CNNs, RNNs, and hybrid models. The paper also discusses the challenges and future research directions in this field. These articles provide a good starting point for a literature survey on patient action detection using machine learning. You can also search for related articles in databases such as PubMed, IEEE Xplore, and ACM Digital Library.

III. METHODOLOGY

The goal of this project is to develop a model that can accurately identify basic human actions, such as chest pain, headache, dizziness, stomach pain, and coma patient movement. The model will be trained on a dataset consisting of videos where individuals are performing different actions, and the label of each video will correspond to the action being performed. The model will learn from this dataset and be able to predict the label of an input, such as a live camera video, that it has not encountered before. In essence, the model will need to learn to distinguish between different human actions based on examples provided in the training dataset. Recognition of human activity solely based on camera feed, without the need for additional sensors. Real-time data of human activity is collected for activity determination and fed into the model for detection, resulting in output indicating the performed activity. The model is easily upgradable and expandable with the ability to add new activities without the need for additional hardware, making it simple to install and train with minimal investment. Unpredicted health imbalance i hospital can be immediately treated and notified so that no harm can be caused to the patients. First level activity of the patient can be easily altered as soon as they record. The Random Forest Classifier is an ensemble learning technique that combines multiple decision trees to create a more accurate and robust model.. The main methodologies used in Random Forest Classifier include: Decision Tree: Random Forest Classifier is based on decision trees, which are flowchart-like structures that recursively split data into subsets based on feature values until reaching leaf nodes where predictions are made. Decision trees can be built using different algorithms, such as ID3, C4.5, CART, and others. Bootstrap Aggregating (Bagging): Random Forest Classifier uses a technique called bootstrapped sampling, where multiple subsets of the original training data are created by randomly selecting samples with replacement. Each subset is used to train a separate decision tree, and the predictions from all the trees are combined to make the final prediction. Random Feature Selection: Random Forest Classifier randomly selects a subset of features from the available features at each split in each decision tree. This random feature selection helps in reducing over fitting and increases diversity among the trees, which contributes to the overall performance of the model. Majority Voting: Random Forest Classifier combines the predictions of all the decision trees in the ensemble using majority voting, where the most frequent class predicted by the trees is chosen as the final prediction. This helps in making robust predictions and reduces the impact of individual decision trees' errors.

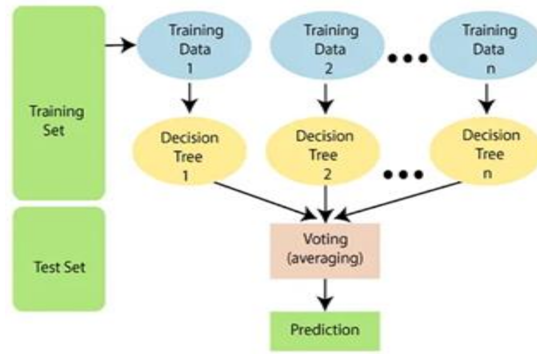

Fig 1: Methodology design

IV. MODELING AND ANALYSIS

a). Pre-processing: In a Random Forest Classifier, a collection of decision trees is trained on the same dataset using a technique called bootstrapped sampling. This means that each tree is trained on a randomly selected subset of the original dataset, with replacement, and the rest of the data is left unused for that particular tree. Additionally, at each split in each tree, only a randomly selected subset of features is considered for determining the best split. Once the individual decision trees are trained, predictions are made by taking a majority vote from all the trees. The class that receives the most votes is predicted as the final output. This ensemble approach helps to reduce over fitting and improve the overall performance of the model.

b). Data Preparation: Collect and prepare the dataset for training. This may involve cleaning the data, handling missing values, and performing feature engineering to extract relevant features from the raw data.

c). Splitting the Dataset: Split the dataset into two or three sets: training set, validation set (optional), and test set. The training set is used to train the model, the validation set is used to tune hyper parameters and optimize the model, and the test set is used to evaluate the final model's performance. Feature Selection (optional): Random Forest Classifier automatically handles feature selection by considering a random subset of features at each split. However, if you have a large number of features and want to manually select a subset of the most important features, you can perform feature selection techniques like feature importance analysis or recursive feature elimination.

d). Building the Random Forest Classifier: Train the Random Forest Classifier on the training set. The classifier creates a forest of decision trees, where each tree is built by randomly selecting a subset of the training data and a subset of features at each split.

e). Hyper parameter Tuning: Optimize the performance of the Random Forest Classifier by tuning its hyper parameters. These hyper parameters may include the number of trees in the forest, the maximum depth of the trees, the minimum number of samples required to split a node, and other relevant parameters.

f). Model Evaluation: The performance of the Random Forest Classifier can be assessed using the validation set. Common evaluation metrics for classification tasks, such as accuracy, precision, recall, F1-score, and area under the receiver operating characteristic (ROC) curve, can be used.

g). Model Optimization (optional): If the model's performance is not satisfactory, you can go back to step 4 and tune the hyper parameters further, or perform additional feature engineering to improve the model's accuracy.

h). Model Testing: Once you are satisfied with the model's performance on the validation set, you can evaluate its performance on the test set to get an unbiased estimate of its generalization performance.

i). Model Deployment: Once the model has undergone training, tuning, and testing, it can be deployed in a production environment to generate predictions on new, unseen data.

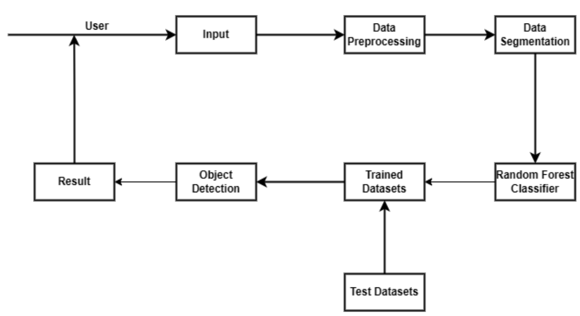

Fig 2 : System Architecture

V. RESULT AND DISCUSSION

In this experimental module we have used the landmark lib for the estimation or to detect the different posture (), initially we are importing required library functions for holistic results in the

Step 1 .cse file which contains data is imported by using the function pd.read_cse is imported as input. A video capture file contains the training dataset is passed to be build module for capturing the feature of the body. For detecting the posture and to recognize the mask, the random forest classifier, logistic regression, ridge classifier, gradientboostingclassfier is used.

Step 2 media pipeline solutions is marked for plotting the posture, the drawingspec will detect the landmarks of face, pose, left hand, right hand of the human body, and to extract the face landmark x,y is used as a utility or an int for consideration of posture recognition. The main detection methodology of different posture will concatenate depending upon the coordinate and feature mapping for the testing videos. And for audio alert of the detected posture we are using the body language class is used for run and wait condition. Where run is the total execution time taken and wait is the ready state where the new process is going to execution state. To train the value we are differentiating 100 datasets into 80 and 20 where 80 is training dataset and 20 is the testing dataset.

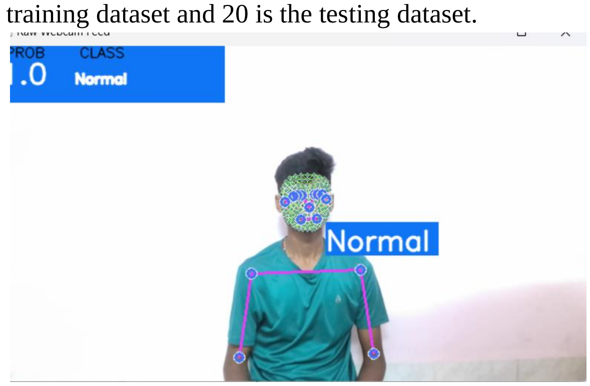

Fig 3: When patient in normal position model detect the action.

Fig 4: When patient suffering from heart pain model detect the action.

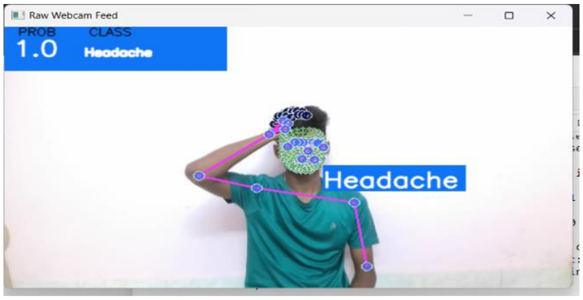

Fig 5: When patient suffering from head ache model detect the action.

Fig 6: When patient suffering from leg pain model detect the action.

VII. CONCLUSION

In conclusion, patient action detection using machine learning (ML) has the potential to greatly improve healthcare outcomes by providing real-time monitoring and analysis of patient activities. ML algorithms can analyze data from various sensors and devices, such as wearable's, smart devices, and cameras, to detect and classify patient actions, such as falls, seizures, or irregular movements. This can enable healthcare providers to promptly respond to critical events, prevent accidents, and provide timely care. One key advantage of using ML for patient action detection is its ability to continuously learn and adapt from new data, allowing for ongoing refinement and improvement of the detection accuracy over time. ML algorithms can also handle large amounts of data, making them capable of detecting subtle patterns and anomalies that may not be easily discernible to human observers. Moreover, ML-based patient action detection can operate in real-time, enabling rapid response and intervention, particularly in time-sensitive situations. However, there are also challenges and considerations to be addressed when implementing ML for patientaction detection. Data privacy and security, ethical considerations, and regulatory compliance must be carefully managed to protect patient confidentiality and ensure compliance with relevant laws and regulations. Additionally, the accuracy and reliability of ML algorithms in real-world settings need to be validated through rigorous testing and evaluation to ensure their safety and effectiveness. In summary, patient action detection using ML has the potential to revolutionize healthcare by providing realtime monitoring and detection of critical patient actions. With proper safeguards and validation, ML-based systems can enhance patient safety, improve healthcare outcomes, and support healthcare providers in delivering timely and appropriate care. Further research and development in this field hold promise for advancing patient care and transforming healthcare practices.

REFERENCES

[1] J. Wang, Y. Chen, S. Hao, X. Peng, L. Hu, ”Deep Learning for Sensor based Activity Recognition: A Survey”, Computer Vision and Pattern Recognition, doi: 10.1016/j.patrec.2018.02.010

[2] 2020 2nd International Conference on Sustainable Technologies for Industry 4.0 (STI), 19-20 December, Dhaka.

[3] Human Activities in Real-time from Smartphone Sensor Data, Accessed on: Jul. 9, 2020. [Online]. Available: http://ahadvisionlab.com/

[4] Hindawi Computational Intelligence and Neuroscience Volume 2022, Article ID 3454167, 10 pages https://doi.org/10.1155/2022/3454167

[5] Human Activity Recognition using Smartphone Article • January 2014 Source: arXiv CITATIONS 24 READS 20,105 3 authors, including: Amin Rasekh Texas A&M University